**Profesionālās izglītības kompetences centrs**

**„Rīgas Valsts tehnikums”**

Izglītības programma: Programmēšana

**KVALIFIKĀCIJAS DARBS**

**Volejbola sacensību datu uzskaites**

**un analītikas sistēma**

Paskaidrojošais raksts 51 lpp.

Audzēknis: Edijs Skudra

Vadītājs: Jeļena Matvejeva

Normu kontrole: Olga Sabanska

**Rīga**

**2021**

**ANOTĀCIJA**

Kvalifikācijas darba ietvaros tika izstrādāts volejbola sacensību portāls datu uzskaitīšanai un datu analītikai. Darba autors ir Rīgas Valsts tehnikuma audzēknis Edijs Skudra. Darba vadītāja ir Rīgas Valsts tehnikuma skolotāja Jeļena Matvejeva.

Kvalifikācijas darbā izstrādātā sistēma ir volejbola sacensību datu uzskaite, kas sevī iekļauj datu analītiku, kurā tiek uzrādīti dati grafikos par efektīvākajiem spēlētājiem, kas reģistrēti lietotnē. Kvalifikācijas darbā galvenais uzsvars tiek likts uz pašu spēles norisi un datiem par to, bet arī datu apstrādi un datu pārveidošanu par grafikiem. Sistēma tika izstrādāta ar programmēšanas valodu JavaScript un datubāžu vadības sistēmu SQLite.

Kvalifikācijas darba paskaidrojošais raksts satur sevī ievadu, lietotāja ceļvedi, uzdevuma nostādni, prasību specifikāciju, uzdevuma risināšanas līdzekļu izvēles pamatojumu, programmatūras produkta modelēšanas un projektēšanas aprakstu, datu struktūru aprakstu, nobeigumu un pielikumus. Ievadā ir aprakstītas aktuālās volejbola nozares problēmas un pamatota vajadzība pēc šāda tipa portāla. Uzdevuma nostādnē ir aprakstīts kvalifikācijas darba galvenais izveidošanas mērķis un tā uzdevumi. Prasību specifikācijā ir aprakstītas sistēmas funkcionālās un nefunkcionālās prasības, kā arī aplūkojama sistēmas izejas un ieejas informācija – dati, ko lietotājs ievada un dati, kuri tiek tam izvadīti. Uzdevuma risināšanas līdzekļu izvēles pamatojumā ir aprakstīts, kāda programmēšanas valoda, teksta redaktors, relāciju datubāžu vadības sistēma tika izmantota sistēmas izstrādē. Programmatūras produkta modelēšanas un projektēšanas aprakstā ir apskatāmas datu plūsmu diagrammas sistēmas funkcionēšanas attēlošanai, ER diagrammas datu bāzes uzbūves un tās pamatprincipu izprašanai, kā arī detalizēti aprakstīta sistēmas arhitektūra. Datu struktūru apraksts satur visu tabulu struktūru un to aprakstu, kā arī tabulu relāciju shēmu. Lietotāja ceļvedis detalizēti attēlo sistēmas informācijas vizuālo izkārtojumu un paskaidro kā pareizi lietot sistēmu. Nobeigumā ir aprakstīts, kas tika izveidots. Pielikums satur ekrānformu piemērus, programmas pirmkodu un datubāzi.

**ANNOTATION**

Within the framework of qualification work there was developed a volleyball competition framework for listing data and data analytics. The Author is Riga State technical school student Edijs Skudra. The supervisor is Riga State technical school teacher Jelena Matvejeva.

Qualification work system is a volleyball competition data accounting which includes data analytics where data are shown in graphs about the most effective players that have been registered. In qualification work main emphasis is the course of the game and data about the game but also data processing and data conversion to graphs. System was made with programming language JavaScript and data database management system SQLite.

Qualification work explanatory consists of introduction, user guide, task statement, requirements specification, task solving funds justification, software product modeling, design description, data structure description, conclusion and attachments. Introduction describes current volleyball industrie problems and justified need of a software like this. Task statement is described in qualifications work with main goals and main tasks. Specification requirements are described in systems functional and non – functional requirements and systems input and output information. Task solving justification describes programming language, text reductor, relation database management system. Software product modeling and design description has description of data flow diagramms for system functioning, ER diagramm database structure and basic principles for understanding. In data structure description contains all table structures and descriptions and relation scheme. User guide is illustrated in detail about system visual informations layout and explains how to use system right. In conclusion there is information about what have I done. Attachment holds screenshot examples and programm source code and database.

**SATURS**

[IEVADS 5](#_Toc22078)

[1. UZDEVUMA NOSTĀDNE 6](#_Toc18745)

[2. PRASĪBU SPECIFIKĀCIA 7](#_Toc11366)

[2.1. Ieejas un izejasinformācijas apraksts 7](#_Toc1331)

[2.1.1. Ieejas informāijas apraksts 7](#_Toc4026)

[2.1.2. Izejas informācijas apraksts 7](#_Toc7566)

[2.2. Funkcionālās prasības 9](#_Toc26441)

[2.3. Nefunkcionālās prasības 9](#_Toc32289)

[3. UZDEVUMU RISINĀŠANA IZVĒLES PAMATOJUMS 11](#_Toc13423)

[4. PROGRAMMATŪRAS PRODKTA MODELĒŠANA UN PROJEKTĒŠANA 13](#_Toc23606)

[4.1. Sistēmas struktūras modelis 13](#_Toc4926)

[*4.1.1* Sistēmas arhitektūra 13](#_Toc30654)

[4.1.2. Sistēmas ER moelis 13](#_Toc12133)

[4.2. Funkcionālās sistēms modelis 16](#_Toc4903)

[4.2.1. Datu plūsmas mdelis 16](#_Toc22833)

[5. DATU STRUKTŪRU APRASTS 21](#_Toc16086)

[6. LIETOTĀJA CEĻVEDIS 28](#_Toc910)

[6.1 Sistēmas prasības aparatūrai un programmatūrai 28](#_Toc15771)

[6.2. Sistēmas instalācija un palaišana 29](#_Toc31554)

[6.3. Programmas apraksts 29](#_Toc12696)

[6.4. Testa piemērs 42](#_Toc26794)

[NOBEIGUMS 49](#_Toc32102)

[INFORMĀCIJAS AVOTI 50](#_Toc30920)

[PIELIKUMI 51](#_Toc25486)

[1. Pieliums 52](#_Toc12136)

[Ekrānform piemēri 52](#_Toc32391)

# IEVADS

Volejbls ir viens no populārākājiem sportaviediem pasaulē. Katru gadu tiek veidotas sacensības volejbolā un šis sporta veids paliek ar vien populārāks visapkārt pasaulei. Ar vien vairāk valstis uzzin par šo sporta veidu un sāk to apgūt.

Pēc padziļinātas interneta resursu izpētes konstatēju, ka Latvijā priekš volejbola datu reģistrēšanas nav interneta tīmekļa portālu kur dati tiek apkopoti un izvadīti pārredzamā veidā. Lielākoties atrodamās tīmekļa vietnes par volejbolu ir novecojušas vai tajās nav iespējams apskatīt spēļu notikumus un norisi. Tās nepiedāvā patīkamu lietotāja pieredzi un funkcionalitāti, kas veicinātu izmantot šo tīmekļa vietni atkārtoti sacensībām nākotnē.

Portāla izveide atrisinās vairākas problēmas. Tiks uzlabota datu glabāšana un to izvade gan pa spēlēm, gan komandām. Būs viegli un ērti reģistrēt datus par spēli. Visas funkcijas būs elementāras, viegli atrodamas un viegli izmantojamas.

Mūsdienās internetam ir ļoti liela nozīme. Šajos laikos visas mājaslapas tiek veidotas tā, lai dati būtu pēc iespējas ātrāk apskatāmi un apstrādājami, un ja notiek kāda ķibele ar lapu to ātri un vienkārši būtu izlabot, lai lietotājam būtu jāgaida pēc iespējas mazāks laiks, līdz šo lapu varēs lietot atkal. Volejbola nozare ir viena no tām, kur tas vēl nav efektīvi un detalizēti pielietots. Šāda lietotne atvieglotu darbu cilvēkiem kuriem nepieciešams datus reģistrēt un apstrādāt tos un to lietotājiem viegli un ātri apskatīt visu nepieciešamo informāciju kuru tie vēlas aplūkot.

# UZDEVUMA NOSTĀDNE

Kvalifikācijas darba uzdevums ir izveidot ērtu volejbola sacensību portālu ar vieglu datu ievadi, kas reģistrē datus augstā līmenī. Sistēmai ir jānodrošina vieglu spēlētāju, komandu un spēļu pievienošanu, vieglu spēles datu ievadi un izvades, un grafikus ar dažādu datu izvadi priekš labākas lietotāja pieredzes un iespējams arī volejbola tirgus.

**Kvalifikācijas darba galvenie mērķi:**

* izveidot ērtu informācijas apskati par spēlētājiem, komandām un spēli;
* izveidot ērtu un ātru informācijas apstrādi priekš datu pievienošanas, labošanas, dzēšanas;
* atvieglot darbu cilvēkiem kuri reģistrē spēles datus;
* izveidot viegli pārlūkojamu statistiku;
* piesaistīt vairāk cilvēku šim sporta veidam;
* Izveidot lapu, kuru varētu lietot pat cilvēks, kurš daudz nesaprot no šī sporta veida.

# PRASĪBU SPECIFIKĀCIA

## 2.1. Ieejas un izejas informācijas apraksts

### 2.1.1. Ieejas informācijas apraksts

* Informācija par komandu, ko ievada administrators un tā sastāv no:
  + komandas nosaukuma, kas ir simbolu virkne, ne īsāka par 3 simboliem un ne garāka par 20 simboliem.
* Informācija par spēlētāju, ko ievada administrators un tā sastāv no:
  + spēlētāja vārda, kas ir simbolu virkne, ne īsāka par 3 simboliem un ne garāka par 20 simboliem;
  + spēlētāja uzvārda, kas ir simbolu virkne, ne īsāka par 3 simboliem un ne garāka par 20 simboliem;
  + spēlētāja vecuma, kas ir vesels skaitlis un kura vērtība ir 1 līdz 99;
  + spēlētāja numura, kas ir vesels skaitlis un kura vērtība ir 1 līdz 20;
  + spēlētāja dzimuma, kuru izvēlas no atlasītāja;
  + rezervista, kas ir opcionāla vērtība.
* Informācija par spēli, ko ievada administrators un tā sastāv no:
  + komandas A, kuru izvēlas no atlasītāja;
  + komandas B, kuru izvēlas no atlasītāja;
  + laika kad plānota, kuru izvēlas no atlasītājiem;
  + ieplānotās vietas, kas ir simbolu virkne, ne īsāka par 3 simboliem un ne garāka par 20 simboliem;
  + kategorijas, kas ir simbolu virkne, ne īsāka par 3 simboliem un ne garāka par 20 simboliem.
* Informācija par spēles notikumiem, ko ievada administrators un tā sastāv no:
  + laika kad izveidots, kas sastāv no datuma un laika;
  + komandas, kuru izvēlas administrators;
  + punkta tipa kuru izvēlas administrators;
  + kļūdas tipa kuru izvēlas administrators;
  + spēlētāja kuru izvēlas administrators.

### Izejas informācijas apraksts

Izejas dati ir no datbāzes izgūtie dati, kā atbilde uz lietotāja pieprasījumu.

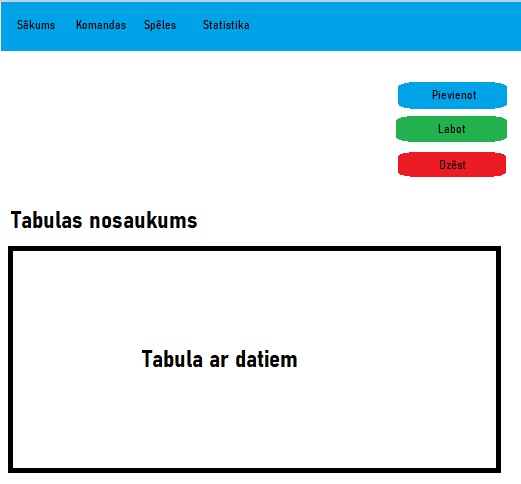
* Informācija par komandu, kas sastāv no:
  + komandas nosaukuma;
  + spēlētāju masīva kurā atrodas visi komandas spēlētāji;
  + spēles notikumi komandai;
  + opcijas - spēle izdzēsta;
  + opcijas - kad spēle izdzēsta, tiek reģistrēts laiks un datums kad tas notika;
  + uzvarētajām spēlēm kur reģistrē vai komanda ir uzvarējusi spēli.
* Informācija par spēlētāju, kas sastāv no:
  + spēlētāja vārda;
  + spēlētāja uzvārda;
  + vecuma;
  + spēlētāja numura;
  + dzimuma id, kas nāk no dzimumu tabulas;
  + komandas id, kas nāk no komandu tabulas;
  + opcijas - rezerves spēlētājs;
  + spēles notikumiemm. Tie ir notikumi no spēles kurus veicis spēlētājs;
  + opcijas - spēlētājs izdzēsts;
  + opcijas - kad spēlētājs izdzēsts, tiek reģistrēts laiks un datums kad tas notika.
* Informācija par spēli, kas sastāv no:
  + plānotās spēles laika un datuma;
  + spēles atrašanās vietas, kur tā notiks;
  + kategorijas id, kas nāk no kategorijas tabulas;
  + spēles statusa, lai zinātu vai spēle ir neiesākta, iesākta vai pabeigta;
  + komandas A id priekš pirmās komandas;
  + komandas B id priekš otrās komandas;
  + spēles notikumiem kur tiek glabāti visi notikumi kuri notikuši šajā spēlē;
  + komandas A punktiem
  + komandas B punktiem
  + komandas A setiem
  + komandas B setiem
  + opcijas – komandas id kura komanda uzvarēja spēli;
  + opcijas – spēle izdzēsta;
  + opcijas - kad spēle izdzēsta, tiek reģistrēts laiks un datums kad tas notika.
* Informācija par spēles notikumiem, kas sastāv no:
  + laika un datuma kad izveidots;
  + tipa id, kas nāk no tipa tabulas, lai zinātu kāds tips piešķirts notikumam, piemēram spēles sākums vai punkts;
  + opcionāls - komandas id, lai zinātu kurai komandai padot notikumu;
  + opcionāls - spēlētāja id, lai zinātu kuram spēlētājam padot notikumu;
  + opcijas – punkta tipa id, lai zinātu kāda tipa punktu piešķirt;
  + opcijas – kļūdas tipa id, lai zinātu kāda tipa kļūdu piešķirt;
  + spēles id, lai zinātu no kuras spēles nāk notikums;
  + punkti komandai A;
  + punkti komandai B.

## Funkcionālās prasības

* Komanda:
  + komandas pievienošana;
  + iespēja veikt komandas nosaukuma maiņu;
  + Iespēja dzēst komandu;
  + visu pievienoto komandas spēļu saraksta apskate;
  + komandas labāko spēlētāju statistika pēc punktiem.
* Spēlētāji:
  + spēlētāja pievienošana;
  + iespēja veikt spēlētāja informācijas labošanu;
  + iespēja dzēst spēlētāju;
  + spēlētāja punktu un kļūdu statistikas apskate.
* Spēle:
  + spēles pievienošana;
  + iespēja dzēst spēli;
  + punkta pievienošana kādam no komandas spēlētājiem;
  + kļūdas pievienošana kādam no komandas spēlētājiem;
  + saraksts ar spēles notikumiem atvērtajai spēlei dilstošā secībā;
  + pēdējā notikuma dzēšana;
  + spēles veikto punktu un kļūdu statistikas apskate;
  + esošo punktu un setu apskate spēlē.
* Punktu statistikas apskate statistikas lapā.
* Sākumlapa:
  + ieplānoto spēļu komandas un plānotais spēles laiks ieplānoto spēļu sarakstā;
  + aktīvo spēļu komandas un spēles rezultāts aktīvo spēļu sarakstā;
  + pabeigto spēļu komandas un spēles rezultāts setos pabeigto spēļu sarakstā.

## Nefunkcionālās prasības

* Prasības procesam:
  + lapai jābūt izstrādātai ar Prisma.io - servera puses bibliotēku un ar CSS3, HTML5 un JavaScript lietotāja puses programmēšanas valodām;
  + izstrādē jāizmanto Chakra UI bibliotēku;
  + jāizmanto React -17.0.2 versijas bibliotēku;
  + lapai jābūt ātram ielādes laikam, nedrīkst pārslogot procesoru.
* Prasības interfeisam:
  + visam jābūt viegli pārredzamam;
  + lapai neizmantot vairāk par trim krāsām kuras nav pārāk uzkrītošas, bet gan vairāk neitrālas;
  + visas lapas veido pēc viena stila;
  + visai informācijai jābūt atdaītai, lai tā nesaplūst ar citu;
  + lapas pamatinformācijai jābūt centrētai (skat. 2.1. att.).



2.1. att. Lapas skice.

# UZDEVUMU RISINĀŠANA IZVĒLES PAMATOJUMS

Kvalifikācijas darba ietvaros tik izmantotas vairākas atklātā pirmkoda skriptu valodas. Viena no tām ir tīmekļu izstrādes programmēšanas valoda JavaScript. Šo programmēšanas valodu izvēlējos, jo tā ir mūsdienīga programmēšanas valoda priekš tīmekļu vietņu izstrādes. Nākotnē par tās turpmāko darbību un regulāru attīstību nav ne mazāko šaubu.

Lai padarītu kodu vieglāk lietojamu un interaktīvo lietotāja interfeisa izveidi vienkāršāku izmantoju React bibliotēku. React ļauj veidot lietotāja interfeisa komponentes un pārlādēt datus liela datu apjoma tīmeļa lapai nepārlādējot to. React izmantošanas galvenā doma ir padarīt lietotni vienkāršu, ātru un mērogojamu (scalable). React ir viena no vadošajām bibliotēkām pasaulē priekš tīmekļu vietņu un mobīlo aplikāciju izstrādes.

Lapas pamata struktūrai tika izmantota hiperteksta iezīmēšanas valoda jeb HTML. Šo iezīmēšanas valodu izvēlējos, jo tā arī ir starptautisks standarts. Kvalifikācijas darbā šo valodu izmantoju, lai strukturētu tīmekļa vietni.

Priekš stila komponentēm izmantoju React komponenšu kopu Chakra UI. Šī bibliotēka paredzēta, lai paātrinātu stila veidošanas procesu ar jau gatavām komponentēm kuras pats vari pārveidot kā vēlies. Tā pieejamība un vienkāršība padara to tik populāru.

Izmantoju arī lapas stilu valodu CSS, priekš darbībām ko nevarēju veikt ar Chakra UI. Kvalifikācijas darbā ši valoda veic stila atdalīšanu no dokumenta satura un ļauj pielāgot dokumenta izskatu konkrēta lietotāja vajadzībām

Lai izveidotu ievades formu izmantoju bibliotēku Formik. Tā ļauj viegli izveidot formu datiem priekš to izvadīšanas. Formik rekomendē izmantot, ja strādā ar React JavaScript.

Lai pievienotu validācijas visiem ievadlaukiem izmantoju bibliotēku yup. Tā ļauj vienkārši un efektīvi pievienot nepieciešamās valdācijas visiem ievadlaukiem. Yup rekomendē izmantot, ja strādā ar Formik.

Izmantoju Prisma.io datubāzes izmantošanai un uzturēšanai. Tā ir servera puses bibliotēka kura palīdz veikt darbības kā datu pievienošanu, labošanu, dzēšanu un lasīšanu drošā veidā.

Izmantoju Hapi servera bibliotēku, jo tā ir viena no populārākajām severa bibliotēkām kuru rekomendē, ja tiek izmantots prisma.io. Hapi izmanto priekš dažādām funkcijām kuras vari veikt ar datiem.

Izmantoju Joi priekš servera datu validācijas, jo pēc statistikas tā ir populāra servera datu validācijas bibliotēka, ja tiek izmantota Hapi bibliotēka.

Izmantoju Node.js priekš servera koda izpildes. Node.js ir atvērtā pirmkoda, starpplatformu JavaScript reāllaika vide, kas ļauj izpildīt JavaScript kodu servera pusē.

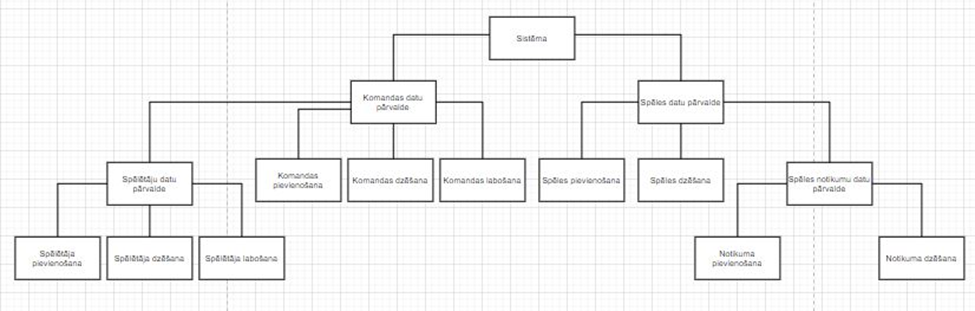
# PROGRAMMATŪRAS PRODKTA MODELĒŠANA UN PROJEKTĒŠANA

## Sistēmas struktūras modelis

### Sistēmas arhitektūra

Volejbola sacensību datu uzskaites un analītikas sistēma sastāv no administratora daļas.

Administratora daļa ir saistīta ar visām sistēmas darbībām. Sistēmas administrātoram būs pieeja visiem skatiem, kuros administrators varēs redzēt visas sistēmas statistiku un pārskatīt, un pārvaldīt visus ierakstus datu bāzē. Pateicoties šiem administrācijas skatiem, administrators vienmēr varēs atrisināt jebkādus tehniskos jautājumus saistībā ar sistēmā pieļautajām kļūdām datu ievadē vai veiktajām darbībām ar datiem, tādejādi veicinot datu precizitāti un pareizību.



4.1. att. Sistēmas arhitektūra

### 4.1.2. Sistēmas ER modelis

Datu bāzes projektēšanā datu kopu un saišu starp tām attēlošanai tika lietota realitāšusaišu diagramma, kas sastāv no divu veidu objektiem – entītēm (loģiskais objekts) un relācijām (attiecības).

Datu bāzes ER modelis sastāv no sekojošām entītijām (skat. 2.pielikumu):

* **„Komandas”** – apraksta komandas sistēmā. Tās atribūtu kopums sevī ietver komandas nosaukumu, uzvarētās, zaudētās spēles un spēles kopā;
* **„Spēlētāji”** – apraksta komandas spēlētājus sistēmā. Tās atribūtu kopums sevī ietver spēlētāja vārdu, uzvārdu, vecumu, dzimumu, spēlētāja numuru un opciju rezervists;
* **„Spēles”** – apraksta volejbola spēli sistēmā. Tās atribūtu kopums sevī ietver divas komandas, atrašanās vietu, datumu, laiku un kategoriju;
* **„Notikumi”** – apraksta spēles notikumus sistēmā. Tās atribūtu kopums sevī ietver spēlētāja notikuma laiku, notikuma tipu un komentāru.

Datu bāzes relācijas arada kā savstarpēji ir savienotas divas vai vairākas entītijas:

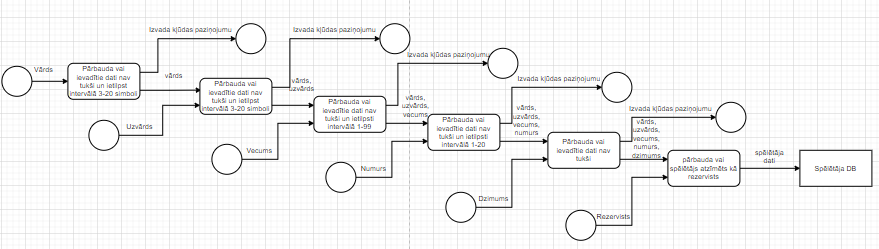
* starp komandām un spēlētājiem attiecības ir daudzi pret daudziem, jo vienai komandai jābūt vairākiem spēlētājiem un viens spēlētājs var spēlēt vairākās komandās;
* starp spēlēm un komandām attiecība ir viens pret daudziem, jo divas komandas izveido vienu spēli, bet vairākas spēles nevar spēlēt tikai viena komanda;
* starp notikumiem un spēlēm attiecība ir viens pret daudziem, jo vienai spēlei var, jābūt ar vairākiem notikumiem, bet viens notikums nevar būt vairākām spēlēm.

## Funkcionālās sistēmas modelis

### 4.2.1. Datu plūsmas mdelis

**1. Spēlētāja pievienošana**

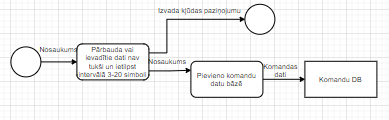
Spēlētājs ir viena no galvenajām sastāvdaļām volejbolā. Bez spēlētāja nebūtu ne komandu ne spēļu. Spēlētāja ievade noris sekojoši. Vispirms administrators ievada spēlētāja vārdu, uzvārdu, vecumu, dzimumu un spēlētāja numuru. Pie katras ievades tiek pārbaudīts vai lauks ir aizpildīts. Pēctam pārbauda vai spēlētāja vārdas un uzvārds nav īsāks par 3 simboliem un garāks par 20 simboliem. Spēlētāja vecums nav mazāks par 1 un lielāks par 99. Spēlētāja numurs nav mazāks par 1 un lielāks par 20. Tiek pārbaudīts vai spēlētājs ticis atzīmēts kā rezervists. Ja spēlētājs netiks atzīmēts kā rezervists tad spēlētājs nebūs rezervists. Ja visi dati netiks aizpildīti izņemot rezervistu, tad spēlētāju nevarēs pievienot.

****

4.2. att. Spēlētāja pievienošanas modelis

**2. Komandas pievienošna**

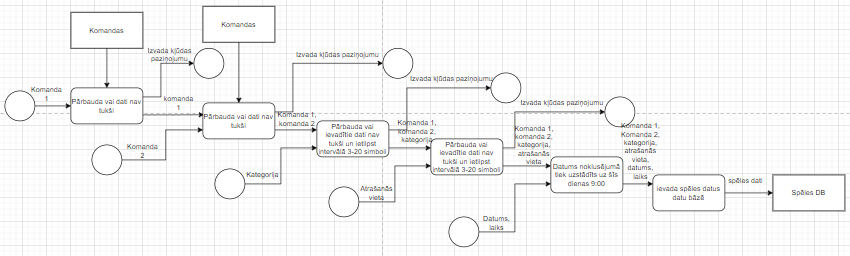
Komandas pievienošana ir svarīga, jo bez komandām nevar tikt veikta spēles pievienošana. Komandas ievade notiek sekojoši. Administrators ievada komandas nosaukumu. Komandas nosaukums nevar būt īsāks par 3 simboliem un garāks par 20 simboliem. Ja nosaukuma lauks nav aizpildīts, tiks izvadīts ka lauks ir tukšs un nevarēs pievienot komandu. Ja informācija ir korekti ievadīta, tad komandas izveide ir iespējama.



4.3. att. Komandas pievienošana

**3. Spēles pievienošana**

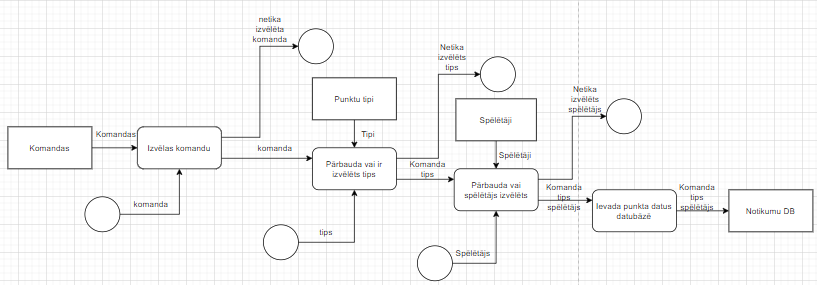
Komandas un spēlētāji ir vajadzīgi tie, kas vajadzīgs priekš spēles izveides. Spēle ir galvenā funkcija šajā portālā. Bez spēles nevarētu plānot spēles, uzsākt tās, reģistrēt spēles datus un no datiem veidot grafikus. Spēles ievade notiek sekojoši. Administrators izvēlas komandas kuras piedalīsies spēlē. Komandas tiek paņemtas no komandu datu bāzes un lietotājam jāizvēlās kuras komandas piedalīsies spēlē. Ja viena vai abas no komandām netika ievadītas tiks paziņots ka dati nav ievadīti pie atbilstošā loga. Pēc komandu ievades tiek ievadīta spēles kategorija. Spēles kategorija ir vismaz 3 simbolus gara un tā nepārsniedz 20 simbolus. Ja kategorija nav aizpildīta tiek izvadīts paziņojums par to. Tālāk tiek pieprasīts ievadīt vietu kur notiks spēle. Vietas nosaukums nevar būt īsāks par 3 simboliem un garāks par 20 simboliem. Ja vieta netika ievadīta tiek paziņots, ka vieta netika ievadīta. Pēc vietas ievades seko datuma un laika ievade. Priekš datuma ievades ir implementēts kalendārs un priekš laika ievades ir izveidoti izvēles lauki. Ja datuma un laika lauki netiek izvēlēti, tad laiks vienmēr būs patreizējās dienas 9:00. Laiku var ievadīt pa stundām un minūtēm. Ja visi dati nebūs aizpildīti izņemot laiku un datumu tad spēli nevarēs pievienot.



4.4. att. Spēles pievienošana

**4. Punkta pievienošana**

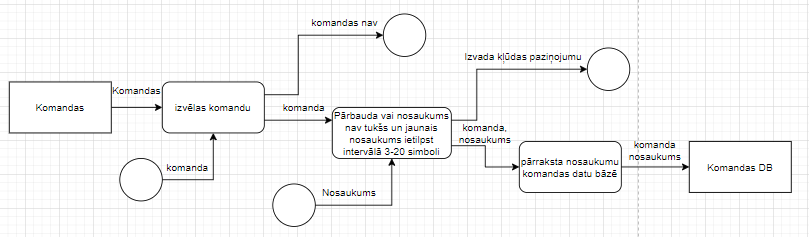
Punkta pievienošanas funkcija ir spēles viena no galvenajām sastāvdaļām. Kļudas pievienošana strādā pēc tāda paša veida. Punkta pievienošana ir tik svarīga, jo pēc tās strādā spēles punkti, spēles seti, notikumu tabula un visi grafiki. Punkta ievade notiek sekojoši. Administrators izvēlas punkta tipu. Ja tips netika izvēlēts tiks paziņots, ka nepieciešams izvēlēties tipu. Kad tips izvēlēts nepieciešams izvēlēties spēlētāju. Ja spēlētājs netika izvēlēts tiks pieprasīts izvēlēties. Ja kādi šiem datiem netika izvēlēti punktu nevarēs pievienot. Ja tiek veikta kļūda, tad punkts iet pretinieku komandai, bet diagramma strādā pēc tāda paša principa.



4.5. att. Punkta pievienošana

**5. Komandas datu labošana**

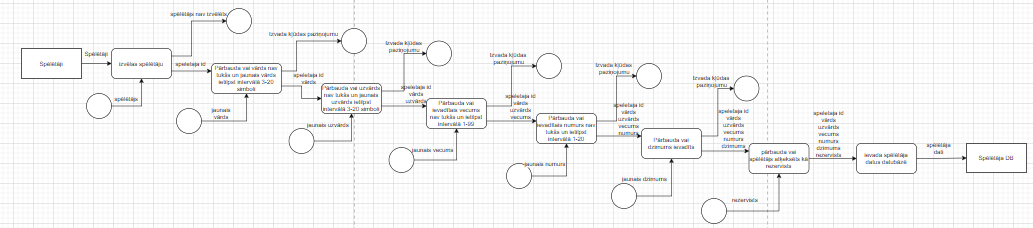
Gadījumā ja komandas nosaukums ir ievadīts nepareizi ir iespēja veikt komandas labošanu. Tā kā komandas pievienošanā ir iespējams pievienot tikai komandas nosaukumu tad komandas nosaukuma labošanā ir iespējams labot tikai komandas nosaukumu. Labojot komandas nosaukumu tam jābūt garākam par 3 simboliem un īsākam par 20 simboliem. Ja komandas nosaukuma lauks ir tukšs, tad komandas nosaukumu nomainīt nevarēs.



4.6. att. Komandas datu labošana

**6. Spēlētāja labošana**

Gadījumā ja spēlētāja ievadītā informācija ir ievadīta nekorekti ir iespēja veikt spēlētāja labošanu. Vispirms tiek izvēlēts spēlētājs kuru vēlies labot. Spēlētāja labošana noris sekojoši. Vispirms administrators ievada spēlētāja vārdu, vārds nevar būt īsāks par 3 simboliem un garāks par 20 simboliem. Ja vārds netika ievadīts tiks izvadīts paziņojums par to ka lauks netika aizpildīts. Tālāk nepieciešams ievadīt uzvārdu. Uzvārds nevar būt īsāks par 3 simboliem un garāks par 20 simboliem. Ja uzvārds nav ticis ievadīts tad tiks izvadīts paziņojums, ka nepieciešams ievadīt uzvārdu. Tālāk tiek pieprasīts ievadīt vecumu. Ja ievadītais vecums būs mazāks par 1 vai lielāks par 99 tiks izvadīts paziņojums par nekorektu vecumu. Ja vecums netiks ievadīts tiks izvadīts paziņojums par to ka vecums netika ievadīts. Tālāk nepieciešams ievadīts spēlētāja numuru. Spēlētāja numurs nevar būt mazāks par 1 un lielāks par 20. Ja spēlētāja numurs netiks ievadīts vai dati būs nekorekti tiks izvadīts paziņojums par to. Pēctam lietotājam nepieciešams izvēlēties dzimumu no esošās izvēles. Ja lietotājs nav izvēlējies dzimumu tad tiks izvadīts paziņojums. Ir iespēja spēlētāju atzīmēt kā rezervistu. Ja spēlētājs netiks atzīmēts kā rezervists, tad viņš automātiski nebūs rezervists. Ja visa informācija ir korekti ievadīta un visi lauki ir aizpildīti, izņemot rezervistu, jo tas nav obligāts, tad būs iespējams izlabot spēlētāju.



4.7. att. Spēlētāja labošana

# DATU STRUKTŪRU APRASTS

Datu bāze sastāv no 11 tabulām, kas satur informāciju par sistēmas spēlētājiem, komandām, spēlēm un spēļu notikumiem. Vairākas tabulas paredzētas priekš jau gataviem datiem piemēram kā punktu vai kļūdu tipiem vai piemēram dzimumiem. Spēles notikumi ir galvenais platformas objekts, kas savienots ar vairākām tabulām un pilda vienu no galvenajām projektā paredzētajām funkcijām. Zemāk tiek attēlota datubāzes datu relācijas shēma (skat. 3.pielikumu).

Tabulā „Team” glabājas informācija par komandu. Informācija par komandas nosaukumu, komandas spēlētājiem, spēles notikumiem, vai tā tikusi izdzēsta un kad, vai komanda tikusi izdzēsta.

5.1 tabula

**„Team” tabulas struktūra**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr**. | **Nosakumus** | **Tips** | **Garums** | **Piezīmes** |
| 1 | id | Autonumber | - | Primārā atslēga |
| 2 | Name | String | 3-20 | Komandas nosaukums |
| 3 | Players | Array | - | Spēlētāju masīvs |
| 4 | GameEvent | Array | - | Spēles notikumi |
| 5 | isDeleted | Boolean |  | Vai komanda ir izdzēsta (opcionāls) |
| 6 | DeletedAt | DateTime |  | Kad komanda tika izdzēsta (opcionāls) |
| 7 | teamAGame | Array | - | Komandas A |
| 8 | teamBGame | Array | - | Komandas B |
| 9 | gamesWon | Array |  | Uzvarētās spēles |

Tabulā **„Player”** glabājas informācija par spēlētājiem komandā. Informācija par spēlētāja vārdu, uzvārdu, vecumu, dzimumu, komandu, rezerves spēlētājs (šī vērtība ir opcionāla, jo ne visi spēlētāji būs rezervisti), spēles notikumiem, vai spēlētājs ticis izdzēsts un kad (būs false jeb nepaties kamēr netiks izdzēsts).

5.2. tabula

**„Player” abulas strutūra**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr**. | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | id | Autonumber | - | Primārā atslēga |
| 2 | firstName | String | 3 līdz 20 | Spēlētāja vārds |
| 3 | lastName | String | 3 līdz 20 | Spēlētāja uzvārds |
| 4 | age | Integer | 1 līdz 99 | Vecums |
| 5 | number | Integer | 1 Līdz 20 | Spēlētāja numurs |
| 6 | gender | Integer |  | Dzimuma id |
| 7 | team | Integer |  | Komandas id |
| 8 | backupPlayer | Boolean |  | Vai spēlētājs ir rezervists |
| 9 | gameEvent | Array | - | Spēles notikumi |
| 10 | isDeleted | Boolean |  | Vai spēlētājs ir izdzēsta (opcionāls) |
| 11 | DeletedAt | DateTime |  | Kad spēlētājs tika izdzēsta (opcionāls) |

Tabulā **„Game”** glabājas infromācija par spēli. Informācija par spēles datumu un laiku, vietu, kategoriju, spēles statusu, komandu A un komandu B, spēles notikumiem, kategoriju, komandu punktiem, komandu setiem, komandu kura uzvarēja (vērtība nebūs tukša līdz brīdim kad kāda no komandām uzvarēs spēli) un vai spēle ir izdzēsta un kad (opcionāls).

5.3. tabula

**„Game” tabulas struktūra**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr**. | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | Id | Autonumber | - | Primārā atslēga |
| 2 | plannedAt | String | 16 | Datums un laiks kad paredzēta spēle |
| 3 | location | String | 3 līdz 20 | Vieta kur notiks spēle |
| 4 | category | Integer | 3 līdz 20 | Kategorijas Id |
| 5 | status | Integer | 3 | Spēle nav sākta / sākusies / beigusies |
| 6 | teamA | Integer | - | Komandas A id |
| 7 | teamB | Integer | - | Komandas B id |
| 8 | events | Array | - | Komandas notikumi |
| 9 | teamAPoints | Integer | 25 | 5 setā 17 | Komandas A punkti |
| 10 | teamBPoints | Integer | 25 | 5 setā 17 | Komandas B punkti |
| 11 | teamASets | Integer | 5 | Komandas A seti |
| 12 | teamBSets | Integer | 5 | Komandas B seti |
| 13 | winingTeam | Integer | - | Komandas id kura uzvarēja (opcionāls) |
| 14 | isDeleted | Boolean |  | Vai spēlētājs ir izdzēsta (opcionāls)) |
| 15 | deletedAt | DateTime |  | Kad spēlētājs tika izdzēsta (opcionāls) |

Tabulā **„GameEvent”** glabājas infromācija par spēles notikumiem. Informācijā par datumu kad izveidots, komandu id, spēlētāju id, punkta/kļūdas tipa id, spēles id, punktu komandai A/B.

5.4. tabula

**„GameEvent” tabulas sruktūra**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr**. | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | Id | Autonumber | - | Primārā atslēga |
| 2 | createdAt | DateTime |  | Kad tika izveidots |
| 3 | type | Integer | - | Tipa id |
| 4 | team | Integer | - | Komandas id |
| 5 | player | Integer | - | Spēlētāja id |
| 6 | pointType | Integer | - | Punkta id (opcionāls) |
| 7 | mistakeType | Integer | - | Kļūdas id (opcionāls) |
| 8 | game | Integer | - | Spēles id |
| 9 | teamAPoints | Integer | 25 | 5 setā 17 | Punkti komandai A |
| 10 | teamBPoints | Integer | 25 | 5 setā 17 | Punkti komandai B |

Tabulā **„Gender”** glabājas informācija par dzimumiem. Paredzēta spēlētāja dzimumiem, ja tiek pievienots vēl kāds.

5.5. tabula

**„Gender” tabulas strutūra**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr**. | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | id | AutoNumber | - | Primārā atslēga |
| 2 | name | String | - | Dzimuma nosaukums |
| 3 | Player | Array | - | Spēlētāja id |

Tabulā **„GameStatus”** glabājas informācija par spēles statusu. Paredzēta spēles statusam.

5.6. tabula

**„GameStats” tabulas truktūra**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr**. | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | id | AutoNumber | - | Primārā atslēga |
| 2 | name | String | - | Statusa nosaukums |
| 3 | Game | Array | - | Spēļu id masīvs |

Tabulā **„GameEventType”** glabājas informācija par spēles notikuma tipu. Paredzēta spēles notikuma tipu.

5.7. tabula

**„GameEvenType” tabuls struktūra**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr**. | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | id | AutoNumber | - | Primārā atslēga |
| 2 | name | String | - | Notikuma tips |
| 3 | GameEvent | Array | - | Spēles notikumu id |

Tabulā **„PointType”** glabājas informācija par punkta tipiem.

5.8. tabula

**„PointTyp” tabulas sruktūra**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr**. | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | id | AutoNumber | - | Primārā atslēga |
| 2 | name | String | - | Punkta tips |
| 3 | GameEvent | Array | - | Spēles notikumu id |

Tabulā **„MistakeType”** glabājas informācija par kļūdu tipiem.

5.9. tabula

**„MistakeType” tabulasstruktūra**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr**. | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | id | AutoNumber | - | Primārā atslēga |
| 2 | name | String | - | Kļūdas tips |
| 3 | GameEvent | Array | - | Spēles notikumu id |

Tabulā **„GameCategory”** glabājas informācija par punkta tipiem.

5.10. tabula

**„GameCatgory” tabula struktūra**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr**. | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | id | AutoNumber | - | Primārā atslēga |
| 2 | name | String | - | Kategorija |
| 3 | Game | Array | - | Spēles id |

# LIETOTĀJA CEĻVEDIS

## .Sistēmas prasības aparatūrai un programmatūrai

Lai sāktu veiksmīgu darbu pie sistēmas, ir nepieciešamas divas lietas:

* lietotāja programmatūra;
* serveris ar uzstādītu programmatūru.

Lietotāja programmatūra ir programmatūra, kas ļauj lietotājam aplūkot uz tīmekļa serveriem esošus dokumentus. Populārākās pārlūkprogrammas ir Google Chrome, Mozilla Firefox, Safari un Microsoft Edge. Sakarā ar to, ka šī sistēma ir paredzēta latviešu valodā un iespējams tā var tikt lietota no vecas pārlūprogrammas, tad lapa iespēju robežās ir pielāgota tā, lai to varētu atvērt arī no vecākas pārlūkprogrammas bez lielām izmaiņām. Uz vecākām pārlūkprogrammām tā var uzvesties savādāk un daži lapas elementi var tikt attēloti savādāk, nekā verot to no jaunākas pārlūkprogrammas. Zemāk ir aplūkojama uzskatāma informācija par lapas atbalstu uz dažādām pārlūkprogrammām attiecīgās operētājsistēmās (skat. 6.1. tabulu).

6.1. Tabula

**Lapas atbalstītās pārlūkprogrammas uz dažādām operētājsistēmām**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Google Chrome | Microsoft Edge | Firefox | Safari |
| Android | Atbalstīts | Atbalstīts | Atbalstīts | - |
| iOS | Atbalstīts | Atbalstīts | - | Atbalstīts |
| Windows | Atbalstīts | Atbalstīts | Atbalstīts | Nav atbalstīts |
| Linux | Atbalstīts | Atbalstīts | Atbalstīts | Nav atbalstīts |

Lapai ir stabils Microsoft Edge atbalsts, kā arī labs atbalsts Google Chrome pārlūkprogrammai uz Windows 10 operētājsistēmas. Uz pārējām pārlūkprogrammām kā Safari un Opera un citām – lapai ir labs atbalsts to jaunākajās versijās. Vecākās versijas nav ieteicams izmantot, jo tajās ir novecojuši standarti un tas nozīmē, ka daži lapas elementi var tikt attēloti nekorekti, jo lapa ir taisīta ar jaunākajām metodēm. Taču pārsvarā lapa ir labi atbalstīta visās iepriekšminēto pārlūkprogrammu versijās, kur katrā var atšķirties dažas nianses, kuras ir visai nenozīmīgas lietotājam un tā pieredzei saskarē ar lapas interfeisu.

Lai veiksmīgi un stabili pārlūkotu lapu, katrai pārlūkprogrammai ir savas noteiktās minimālās prasības attiecībā uz programmatūru un lietotāja datora specifikācijām. Minimālās prasības katrai pārlūkprogrammai par datora specifikācijām var atrast, meklējot kādā no pārlūkiem.

## Sistēmas instalācija un palaišana

Sistēmai ir divu veidu palaišana. Pirmā no tām ir lietotāja puses palaišana. Lietotāja puses palaišana nav nekas vairāk par vienkāršu ieiešanu lapā ar pārlūkprogrammas starpniecību. Viss, kas lietotājam ir jāizdara, ir jāievada pārlūkprogrammā lapas saite un tālāk lietotājs var brīvi lietot šo lapu.

Otrais palaišanas veids ir servera pusē. Šis palaišanas veids aktivizē pašu sistēmas darbību un sagatavo lapu publikai, līdz ar to atļaujot lietotājam to izmantot un brīvi aplūkot. Sistēmas sagatavošana publikai norit sekojoši:

* Sākumā klienta pusē uztaisa būvējumu ar komandu - npm run build;
* Build mapes saturu uzliek uz servera (ngnix, docker, ...);
* Servera puses kodu uzliek uz servera un startē kodu ar komandu – npm start.

## Programmas apraksts

Kad lietotājs ieiet sākumlapā, kur atrodas, kur atrodas sistēma, viņš redz izvēlni. Tālāk zem šīs izvēlnes atrodas spēles drīzumā, aktīvās spēles un pabeigtās spēles.

**Galvenā izvēlne**

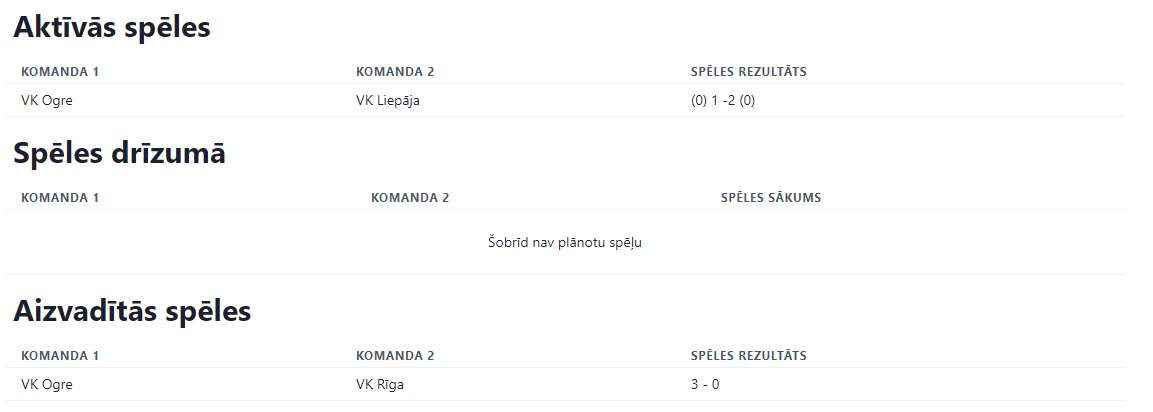
Šī izvēlne atrodas pašā lapas augšā kreisajā malā (skat. 6.1. att.). Tā sevī iekļauj saites uz lapām – sākums, komandas, spēles un statistika. Pirmā saite ir „Sākums”, kura aizved uz tīmekļa sākumlapu, kurā redzamas „Aktīvās spēles”, „Aizvadītās spēles” un „Spēles drīzumā” tabulas kuras sevī iekļauj saites, kuras aizved uz spēlēm. Otrā saite ir „Komandas”, kura aizved uz komandu sarakstu, kur iespējams pievienot jaunu komandu. Trešā saite „Spēles” aizved uz spēļu sarakstu, kur iespējams ieplānot jaunu spēli. Ceturtā saite „Statistika” aizved uz lapu kurā atrodas dažādi grafiki ar statistikas datiem par spēlētājiem.

C:\Users\Laptop\AppData\Local\Microsoft\Windows\INetCache\Content.Word\menu.png

6.1. att. Lapas galeā izvēlne

**Spēles pēc statusa**

Spēles pēc statusa atrodas sākumlapas centrā (skat. 6.2. att.). Spēles ir sadalītās pēc statusa trīs tabulās. Aktīvājās spēlēs, spēlēs drīzumā un aizvadītajās spēlēs. Aktīvo spēļu tabulā ir redzami abu komandu nosaukumi un spēles rezultāts kurā iekļauj gan patreizējos punktus, gan setus katrai komandai. Tabulā spēles drīzumā ir redzami abu komandu nosaukumi un paredzētais spēles sākums kurā tiek attēlots datums un laiks. Trešajā tabulā tiek izvadītas aizvadītās spēles kur ir redzami abu komandu nosaukumi un spēles rezultāts kurā tiek attēloti seti. Tabulas strādā kā saites un izvēloties spēli ar peles klikšķi uz tabulas datiem saite aizved uz spēli kura tika izvēlēta. Ja tabulā datu nav, tad izvadīts teksts - šobrīd nav šādu datu - atkarīgs no tabulas.

6.2. att. Spēles pēcstatusa

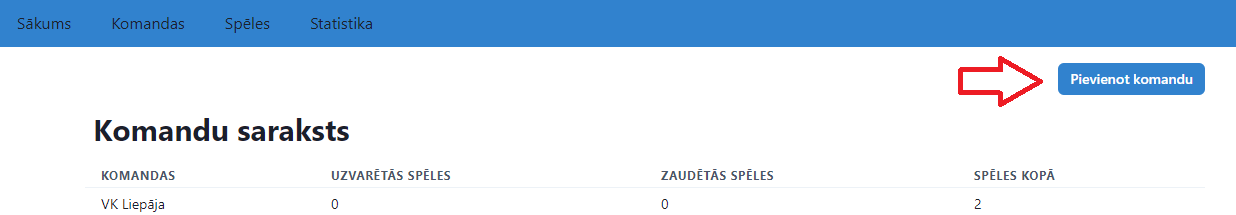
**Komandu lapa**

Komandu lapa atrodas saites Komandas lapas centrā (skat. 6.3. att.). Šajā lapā ir iespēja pievienot jaunu komandu un ir pieejama tabula komandu saraksts. Komandu saraka tabulā ir redzams komandas nosaukums, uzvarētās spēles, zaudētās spēles un spēles kopā. Šī tabula strādā kā saite. Izvēloties komandu un noklikšķinot uz tās saite, tā aizved uz izvēlētās komandas lapu.

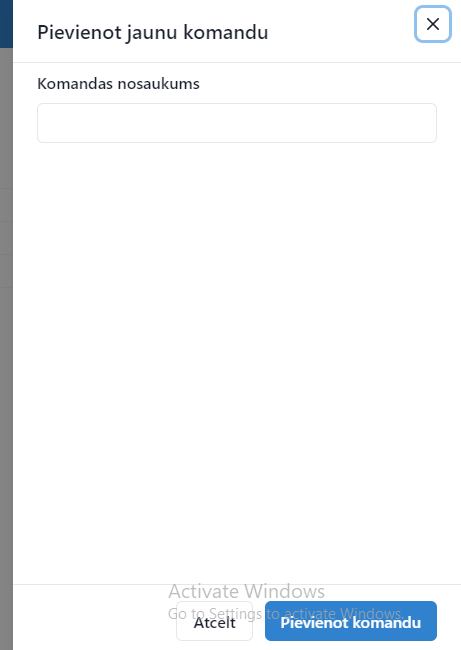
6.3. att. Komandu lapa

**Komandas pievienošana**

Komandas pievienošanas poga atrodas starp galveno izvēlni un komandu sarakstu labajā komandu lapas stūrī (skat 6.4. att.). Komandas pievienošanas formā (skat 6.5. att.) ir atrodams ievadlauks, kuru nepieciešams aizpildīt, lai pievienotu komandu. Lauku validācija notiek automātiski, ja uz lauka tiek uzspiests un tiek ievadīti vai neievadīti dati. Apakšā ir atrodama poga „Pievienot komandu”, kura pieprasa, lai lauki būtu aizpildīti gadījumam, lai nevarētu pievienot komandu bez nosaukuma.



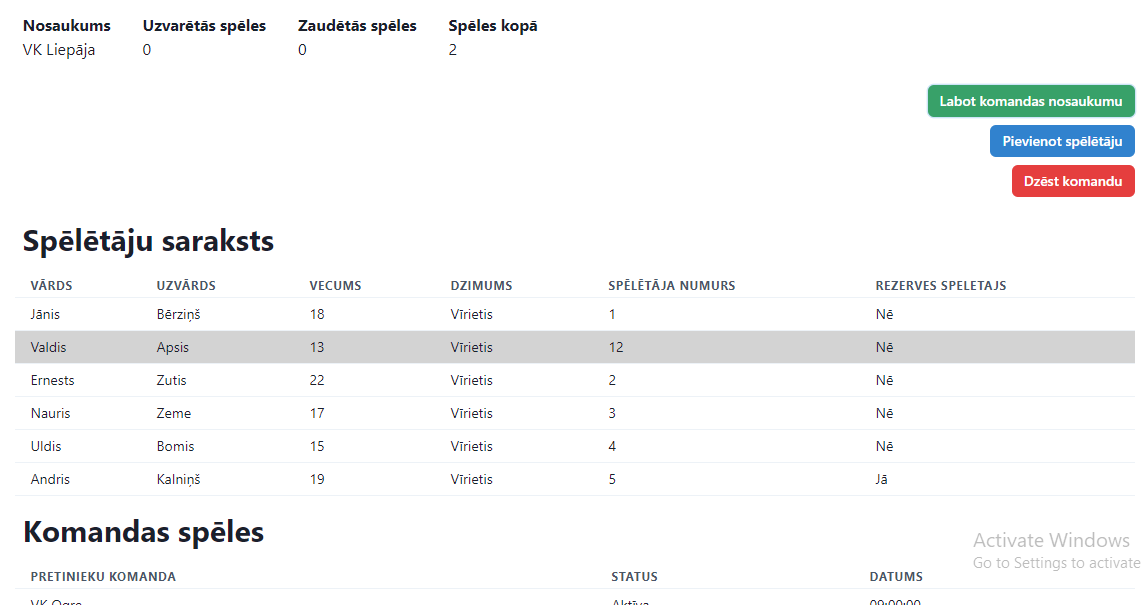
6.4.att. Komandu pievenošana

****Komandas pievienošanas atvilkne kurā nepieciešams ievadīt komandas nosaukumu. Nosaukums nedrīkst būt tukšs un tā simbolu intervāls ir 3-20 simboli (skat 6.5. att.).

* 1. att. Komadas pievienošanas atvilkne

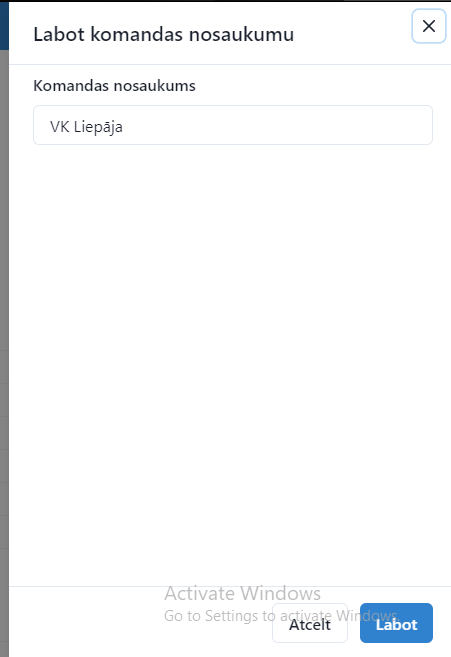
**Komandas lapa**

Nospiežot uz saites „Komandu saraksts”, tabulas saites lietotājs tiek novirzīts uz izvēlētās komandas lapu (skat. 6.6. att.). Komandas lapā tiek attēlota tikai izvēlētās komandas informācija.



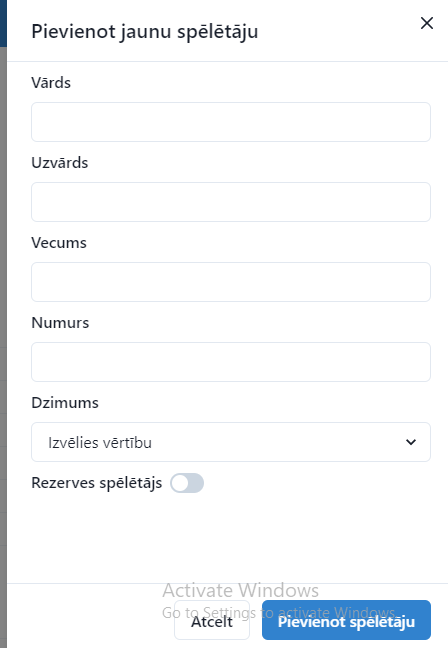
* 1. att. Komandas lapa

Nospiežot pogu „Labot komandas nosaukumu” no labās puses izlec atvilkne (skat. 6.7. att.), kurā ir atrodams ievadlauks, kuru nepieciešams aizpildīt, lai nomainītu komandas nosaukumu. Ievadlaukā būs redzams patreizējais komandas nosaukums. Lauku validācija notiek automātiski, ja uz lauka tiek ievadīti dati. Apakšā labajā stūrī ir atrodama poga „Labot”, kuru nospiežot tiek pārbaudīts vai ievadlauks nav tukšs.



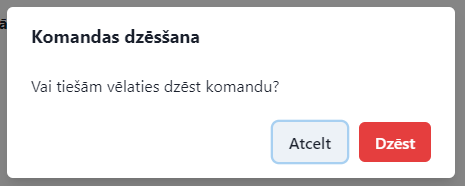
* 1. att. Komandas nosaukuma labošana

Nospiežot pogu „Pievienot spēlētāju” no labās puses izlec atvilkne (skat. 6.8. att.), kurā ir atrodami ievadlauki kurus nepieciešams aizpildīt, lai komandai pievienotu spēlētāju. Lauku validācija notiek automātiski, ja uz lauka tiek ievadīti dati. Apakšā labajā stūrī ir atrodama poga „Pievienot spēlētāju”, kuru nospiežot tiek pārbaudīts vai ievadlauks nav tukšs.



* 1. att. Spēlētāja pieienošana

Nospiežot pogu „Dzēst komandu” izlec, brīdinājuma logs (skat. 6.9 att.) kā pārbaude. Lai izdzēstu komandu jānospiež poga „Dzēst”. Pēc komandas izdzēšanas atgriezt to būs iespējams, izmainot komandas datus datu bāzē.



6.9.att. Komandas dzēšana

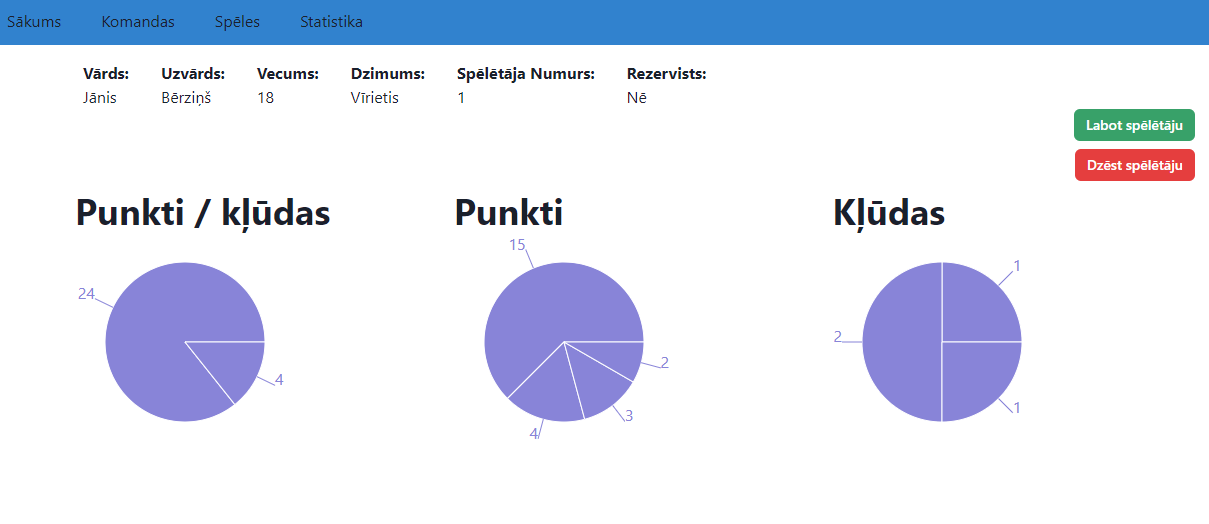
Zem pogas „Dzēst spēli” (skat. 6.6. att.) ir saraksts ar komandas spēlētājiem. Tabulā redzama visa informācija par spēlētāju kuru ievada pievienojot spēlētāju (skat. 6.10. att.). Saraksts strādā kā saite, nospiežot uz kāda spēlētāja saite aizved uz lapu par spēlētāju.



6.10. att. Spēlētāju araksts

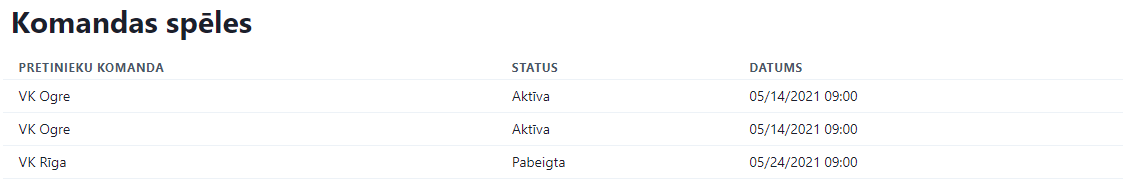
**Spēlētāja lapa**

Spēlētāja lapā ir redzama informācija par spēlētāju, kura tika pievienota, izveidojot spēlētāju, ir iespēja labot un dzēst spēlētāju. Zem pogas dzēst spēlētāju lapas kreisajā pusē ir atrodamas trīs sektoru diagrammas. Pirmajā diagrammā ir redzama punktu un kļūdu attiecība pa visām spēlēm. Otrajā sektoru diagrammā ir redzams visu iegūto punktu tipu daudzums pa visām spēlēm, bet trešajā diagrammā ir redzams visu paveikto kļūdu tipu daudzums pa visām spēlēm (skat. 6.11. att.).



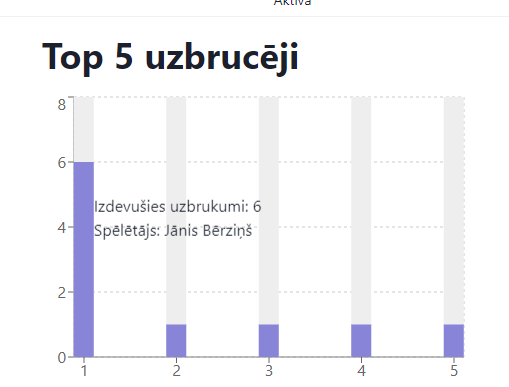
6.11. att. Spēlētāja apa

Zem spēlētāju saraksta atrodas tabula „Komandas spēles” (skat. 6.12. att.). Saraksts strādā kā saite un izvēloties komandas spēli saite aizved uz izvēlēto spēli. Sarakstā ir informācija par pretinieku komandu, spēles statusu un datumu. Statuss norāda, vai spēle nav sākusies, ir aktīva vai ir pabeigta.



6.12. Komandas spēles

Zem „Komandas spēles” tabulas atrodas grafiki ar komandas spēlētāju statistiku (skat. 6.13. att.). Pirmais no grafikiem ir top 5 uzbrucēji no komandas. Stabiņos ir informācija par punkta tipu un spēlētāja vārdu un uzvārdu. Dati tiek apkopoti no visām spēlēm, kurās šie spēlētāji ir piedalījušies.



6.13. att. Komandas satistika

**Spēļu lapa**

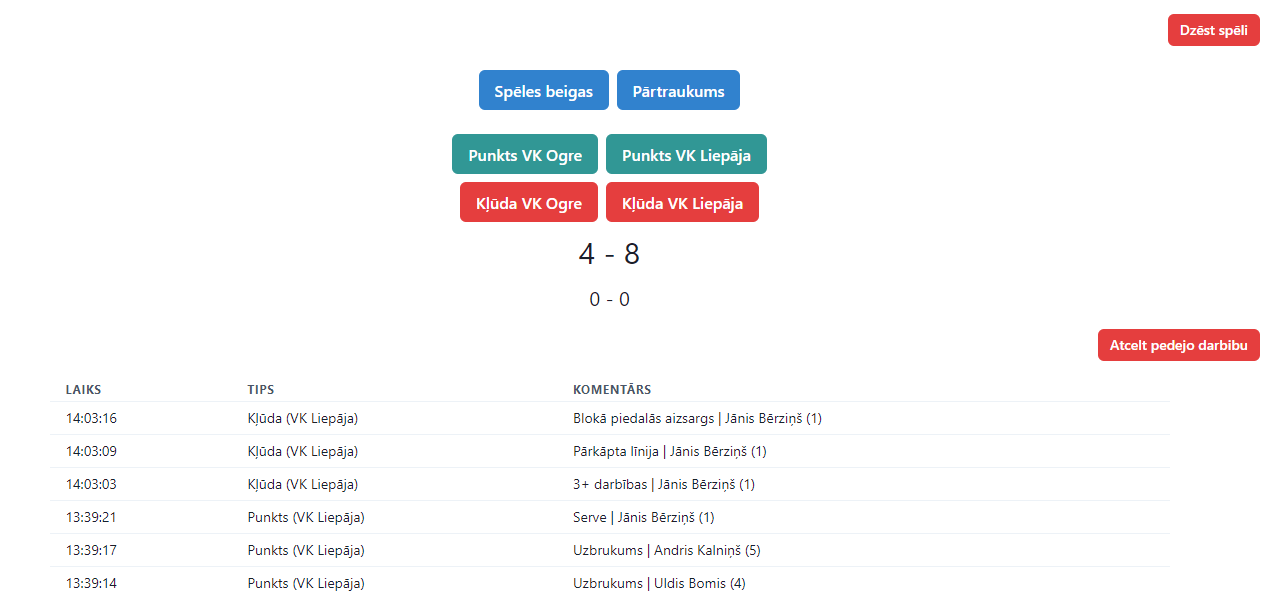
Spēļu lapā ir iespēja pievienot spēli un apskatīt spēļu sarakstu (skat. 6.14. att.). Spēli var pievienot tāpat kā spēlētāju vai komandu. Spēļu sarakstā ir redzama informācija par abu komandu nosaukumiem, spēles kategoriju, norises vietu un spēles paredzēto sākumu. Spēļu saraksts strādā kā saite un, izvēloties kādu no spēlēm, saite aizvedīs uz spēles lapu.



6.14. att. Spēļu lapa

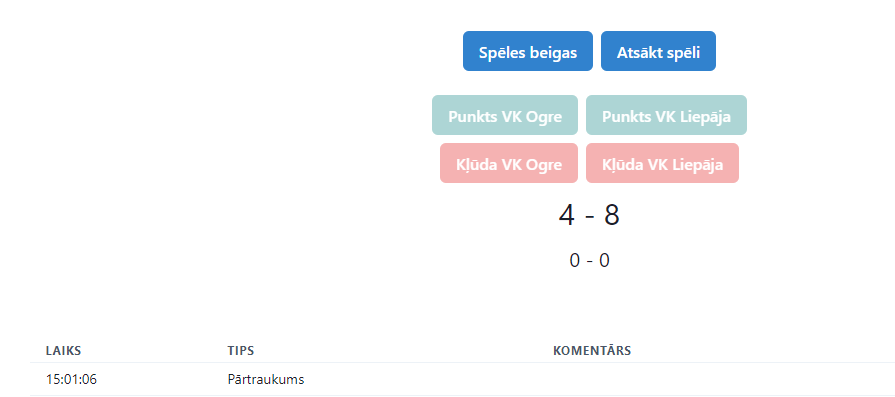
**Spēles lapa**

Spēles lapa ir viena no galvenajām lapām šajā portālā. Spēles lapā norisinās spēļu notikumu reģistrēšana tālākām darbībām (skat. 6.15. att.). Šajā lapā ir iespēja dzēst spēli tāpat kā komandu un spēlētāju.



6.15. att. Spēles lapa

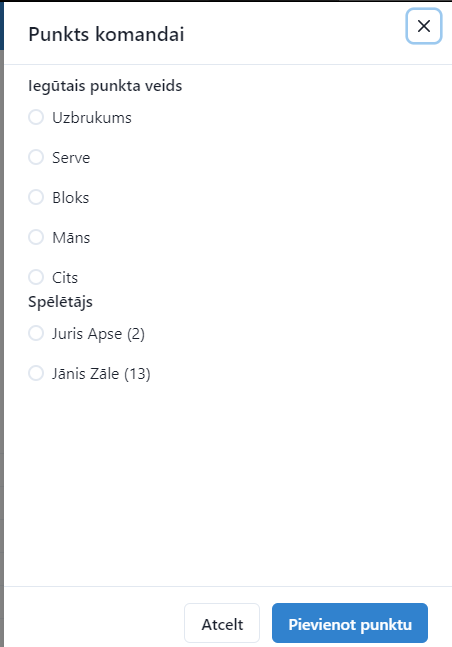
Zem pogas „Dzēst spēli” centrā ir atrodamas pogas „Spēles sākums” vai „Spēles beigas”, atkarīgs no tā vai spēle sākusies vai beigusies. Blakus atrodamas pogas „Pārtraukums” un „Atsākt spēli” (skat. 6.16. att.). Ja tiek veikta kāda no darbībām visi dati tiek reģistrēti un izvadīti zemāk tabulā dilstošā secībā. Gadījumā ja spēle nav sākusies, vai ir beigusies, vai ir pārtraukums, nav iespējams pievienot punktu, jo pogas uz to brīdi tiek deaktivizētas.



6.16. att. Pārtraukums

**Punkta pievienošana**

Punkta vai kļūdas pievienošana notiek pēc viena principa. Nospiežot kādu no tam paredzētajām pogām (skat. 6.15. att.) no labās puses izlec atvilkne ar punktu vai kļūdu tipiem atkarībā no nospiestās pogas un spēlētāji atkarībā kurai komandai tiks piešķirts (skat. 6.17. att.). Pievienojot punktos, tiks reģistrēts laiks, pie tipa tiks pateikts, ka tas ir punkts un komentārā uzrādīs punkta tipu, spēlētāja vārdu uzvārdu un spēlētāja numuru.



6.17. att. Punkta pievienošana

Zem kļūdu pogām ir redzami patreizējie spēles rezultāti - punkti un seti (skat. 6.15. att.).

Starp spēles rezultātiem un spēles notikumu tabulas ir poga atcelt pēdējo darbību (skat. 6.15. att.). Nospiežot šo pogu tiek izdzēsts pēdējais notikums no tabulas un gadījumā ja tas ir punkts vai sets tad tiek izlabots nepieciešamais. Gadījumos kad ievadot vienu notikumu notiek divi vai trīs tiek atcelti visi trīs notikumi. Piemēram tiek reģistrēts punkts. Tas ir pēdējais sets un sekojoši arī spēles beigas.

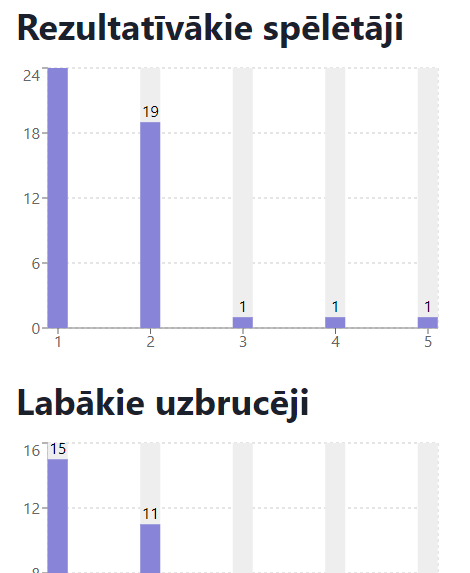
Zem notikumu tabulas ir divi spēles grafiki (skat. 6.18. att.). Pirmajā grafikā tiek attēloti pieci rezultatīvākie spēlētāji šajā spēlē. Otrajā pieci kļūdainākie spēlētāji spēlē. No grafika var nolasīt spēlētāju kurš attēlots uz katra stabiņa un punktu vai kļūdu skaitu.



6.18. att. Spēles spēlētāju statistika

**Statistikas lapa**

Statistikas lapa sastāv tikai no grafikiem (skat. 6.19. att.). Grafiki veidoti no visu spēļu notikumiem. Grafikos attēloti pieci labākie spēlētāji katrā jomā. No grafika stabiņa var nolasīt spēlētāja vārdu, uzvārdu, komandu un punktu skaitu.

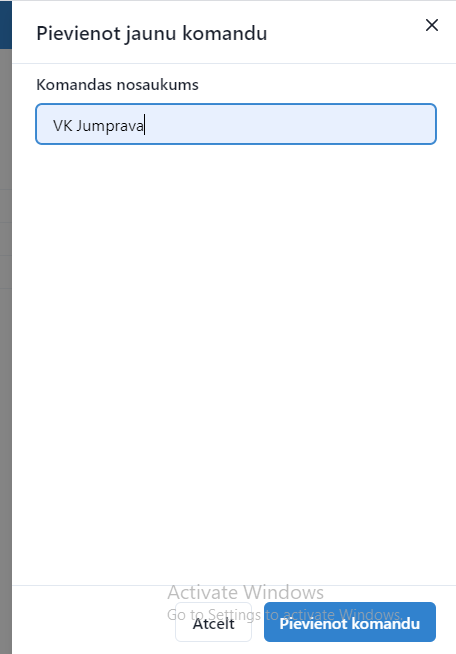


6.19. att. Statistikas lapa

## 6.4. Testa piemērs

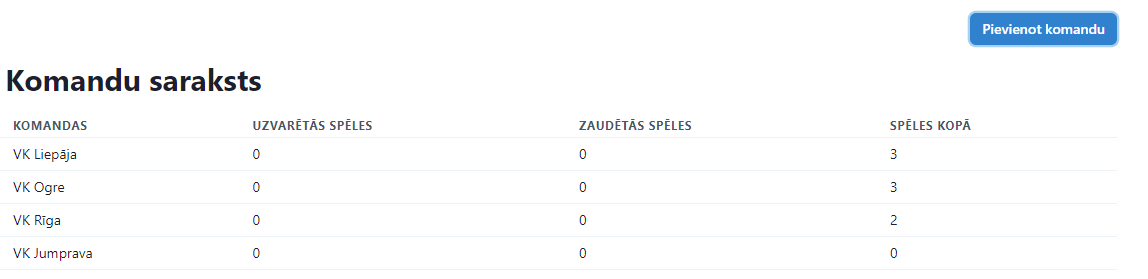
Teta piemēra nolūks ir parādīt visas datu pievienošanas iespējas – spēlētāja, komandas, spēles un punkta pievienošanu.

Pirmais solis ir atvērt lapu. Kad lapa atvērta jādodas uz „Komandas” saiti kas atrodas galvenajā izvēlnē lapas augšā, kreisajā stūrī (skat. 6.20 att.). Kad atvērta komandu ir nepieciešams izveidot komandu. Labajā stūrī virs komandu saraksta ir poga „Pievienot komandu”. Nospiežot uz šis pogas izleks atvilkne no labās puses. Aizpildot komandas nosaukuma logu atvilknē jānospiež „Pievienot komandu” un komanda tiks pievienota (skat. 6.21 att.).



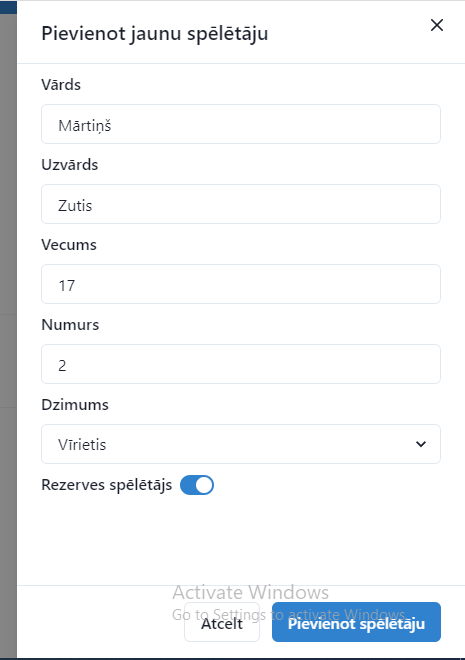
6.20. Komandas pievenošana

Ja komandas izveide bijusi veiksmīga, komanda tiek pievienota komandu sarakstam (skat. 6.21 att.).



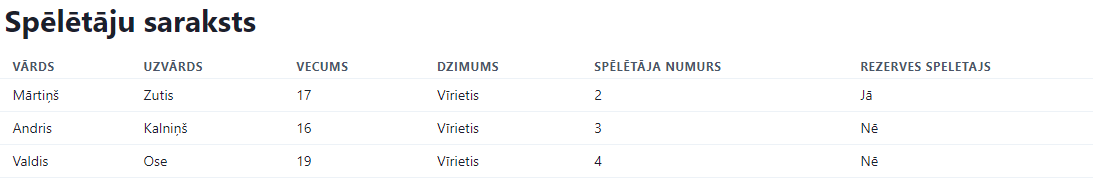
6.21. att. Komanda komandu sarakstā.

Tālāk nepieciešams pievienot spēlētājus. Lai pievienotu spēlētājus komandu sarakstā nepieciešams izvēlēties komandu un atvērt komandu nospiežot uz tās komandu sarakstā (skat. 6.21 att.). Kad komanda ir atvērta labajā stūrī ir poga „Pievienot spēlētāju”. Nospiežot uz šīs pogas no labās puses izleks atvilkne kurā jāaizpilda lauki par spēlētāju. Kad lauki aizpildīti ar nepieciešamo informāciju, lai pievienotu spēlētāju nepieciešams nospiest uz pogas „Pievienot spēlētāju” (skat. 6.22 att.).



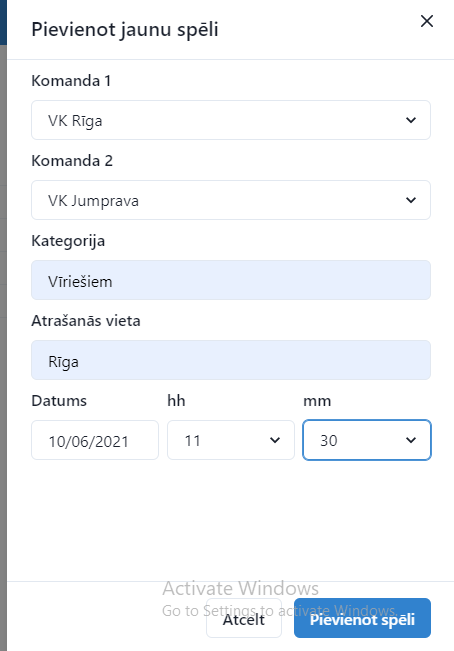
6.22. att. Spēlētāj pievienošana

Ja spēlētāja izveide bijusi veiksmīga, spēlētājs tiek pievienots spēlētāju sarakstam (skat. 6.23 att.).



6.23. att. Spēlētāju saraksts

Kad spēlētāji pievienoti nepieciešams doties uz spēļu sarakstu. Spēļu sarakstu iespējams atrast galvenajā izvēlnē (skat. 6.1. att.). Kad spēļu saraksts atvērts labajā pusē virs spēļu saraksta ir poga „Pievienot spēli” (skat. 6.14. att.). Nospiežot pogu pievienot spēli tiek atvērta atvilkne labajā pusē. Aizpildot informāciju par spēli nepieciešams nospiest „Pievienot spēli”, lai pievienotu spēli spēļu sarakstam (skat. 6.24. att.).



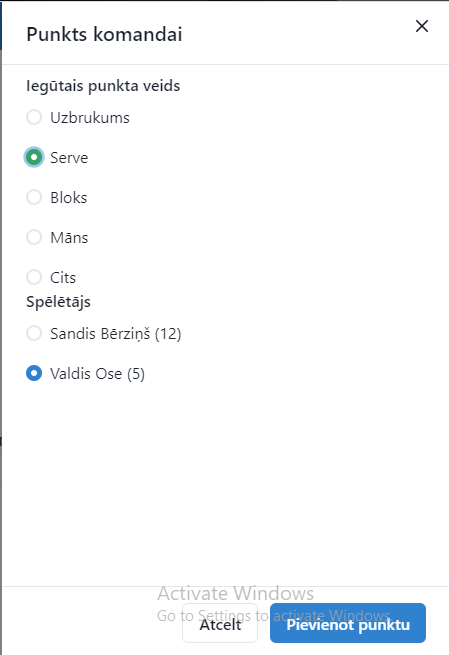
6.24. att. Spēlētāja pievienošana

Ja spēles izveide bijusi veiksmīga, spēle tiek pievienota spēļu sarakstam (skat. 6.25. att.).



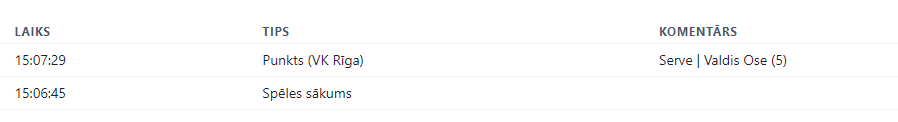
6.25. att. Spēļu saraksts

Nākamais solis ir atvērt spēli. Spēli var atvērt izvēloties spēli spēļu sarakstā un nospiežot uz izvēlētās spēles. Kad spēles lapa atvērta pa vidu ekrānam ir poga „Sākt spēli”. Poga sākt spēli pievienos notikumu sarakstam. Kad spēle ir uzsākta tiek aktivizētas pogas punktiem, kļudām un pārtraukumam. Nospiežot pogu „Punkts Rīga” (skat. 6.15. att.) no labās puses izlec atvilkne ar punktu tipiem un komandas spēlētājiem (skat. 6.15. att.). Nepieciešams izvēlēties vienu punkta tipu un vienu spēlētāju kuram piešķirs šo punktu. Kad tips un spēlētājs izveidots jāspiež poga „Pievienot punktu”(skat. 6.26. att.).



6.26. att. Punkta pievienošana

Ja punkta izveide bijusi veiksmīga, punkts tiek pievienota spēles notikumu sarakstam (skat. 6.27. att.).



6.27. att. Spēles notikumu saraksts

# NOBEIGUMS

Kvalifikāias darbā izvirzītaie mērķi ir sasniegti. Ir izveidots portāls, kas ir viegli lietojams, saprotams ar ērtu informācijas apstrādi un statistiku.

Darbā izvirzītie uzdevumi ir Latvijas mērogā - paredzēti sacensību datu reģistrēšanai ar dažādu statistikas izvadi priekš datu reģistrētāju darba atvieglošanas un vieglākai datu aplūkošanai. Portāls paredzēts, lai veicinātu datu reģistrātorus uzglabāt spēļu datus šajā portālā, atvieglotu tiem darbu, uzglabātu datus ar papildus informāciju par tiem un ja nepieciešams ātri un viegli aplūkot nepieciešamos datus. Šis portāls palīdzētu atīstīt volejbola datu reģistrāciju un ievērot spēlētājus ar potenciālu spēlēt augstāka līmeņa komandās.

Darba sākumā galvenais mērķis bija realizēt ērtu servera un lietotāja puses sistēmu, lai laika gaitā varētu implementēt jaunas funkcijas bez sarežģījumiem īsā laika posmā. Šī iemesla dēļ, liela uzmanība tika pievērsta dinamisku funkciju izstrādei kuras laika gaitā varētu uzlabot pēc nepieciešamības.

Izstrādājot darbu, secināju, ka esmu ieguvis stingru pamatu savā, zināšanām un uz šīs lapas pamata varētu sākt jebkādas lapas izstrādi un viegli to pielāgot tās vajadzībām. Iegūtās zināšanas no ši darba man lika saprast to, portālu izstrāde nav vienkārša un tehnoloģijas mainās strauji.

# INFORMĀCIJAS AVOTI

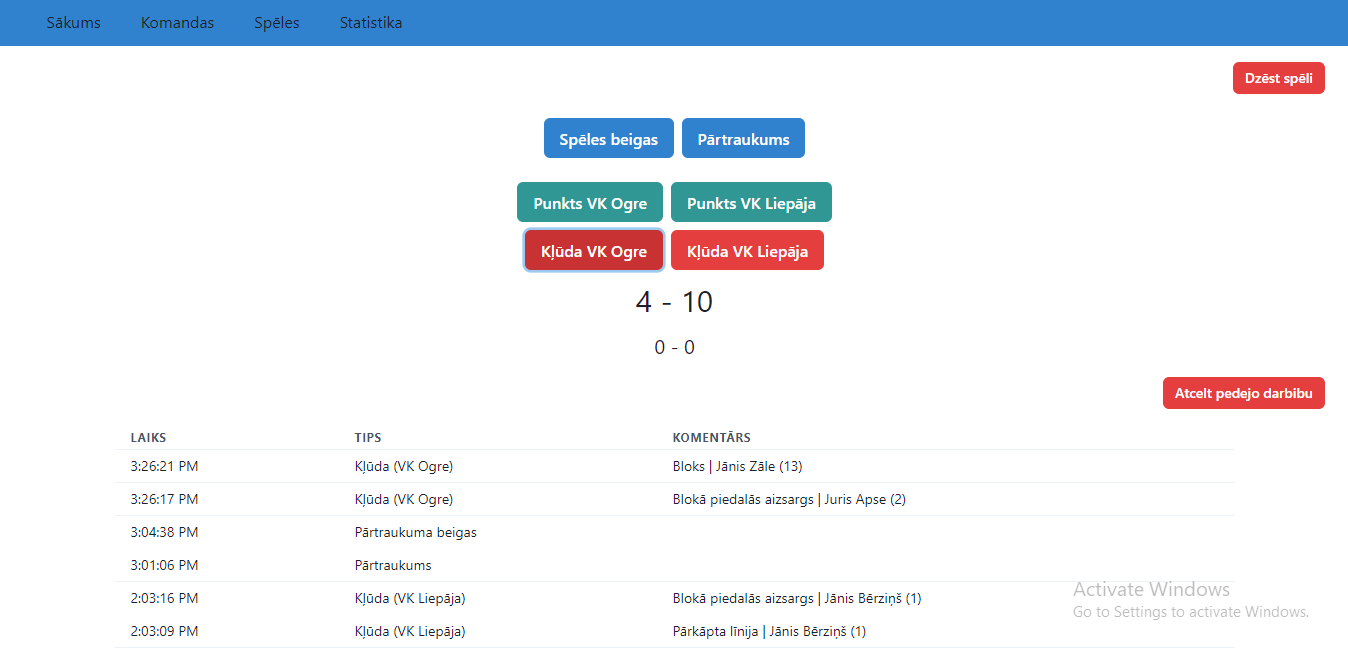
* puvu HTML, CSS, JavaScript <https://www.freecodecamp.org/fcccb371452-9a78-40da-b3f2-756ac85652d1> un <https://www.codewars.com/users/Edijs1>
* Prisma <https://www.prisma.io/>
* Hapi, Joi <https://hapi.dev/>
* React - <https://reactjs.org/>
* Chakra UI <https://chakra-ui.com/>
* Formik, yup - <https://formik.org/docs/guides/validation>
* Rechart <https://recharts.org/en-US/>
* Node.js <https://nodejs.org/en/>

# PIELIKUMI

1. Pielikums

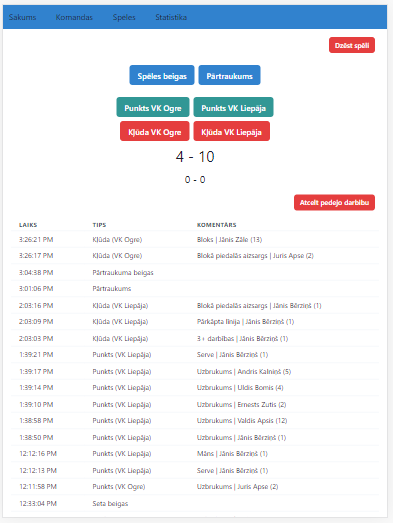
## Ekrānform piemēri

**Lapas izskats datorā (skat. 1.pielikuma 1. att.).**



1. pielikuma 1. att pēles lapas izskats datorā

**Lapas izskats planšetē (skat. 1. pielikuma 2. att.).**



1. pielikuma 2. att. Lpas izskats planšetē

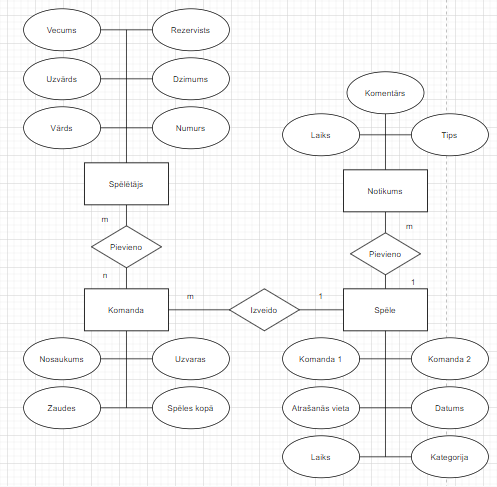
**Lapas izskats mobilajā telefonā (skat. 1. pielikuma 3. att.).**

1. pielikuma 3. att. Lapas izskats planšetē



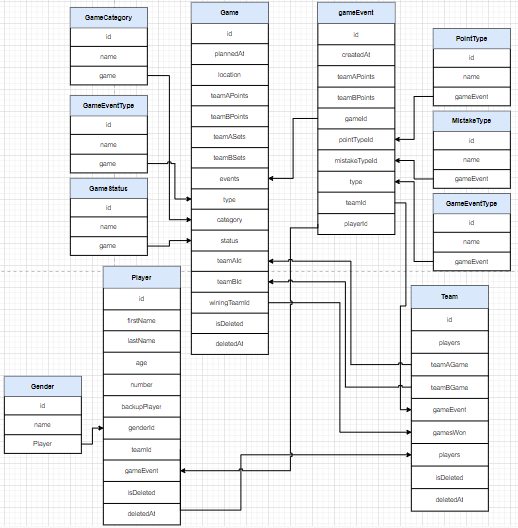
2. pielikums

**ER diagramma**



3. pielikums

**Datu relāciju shēma**

****